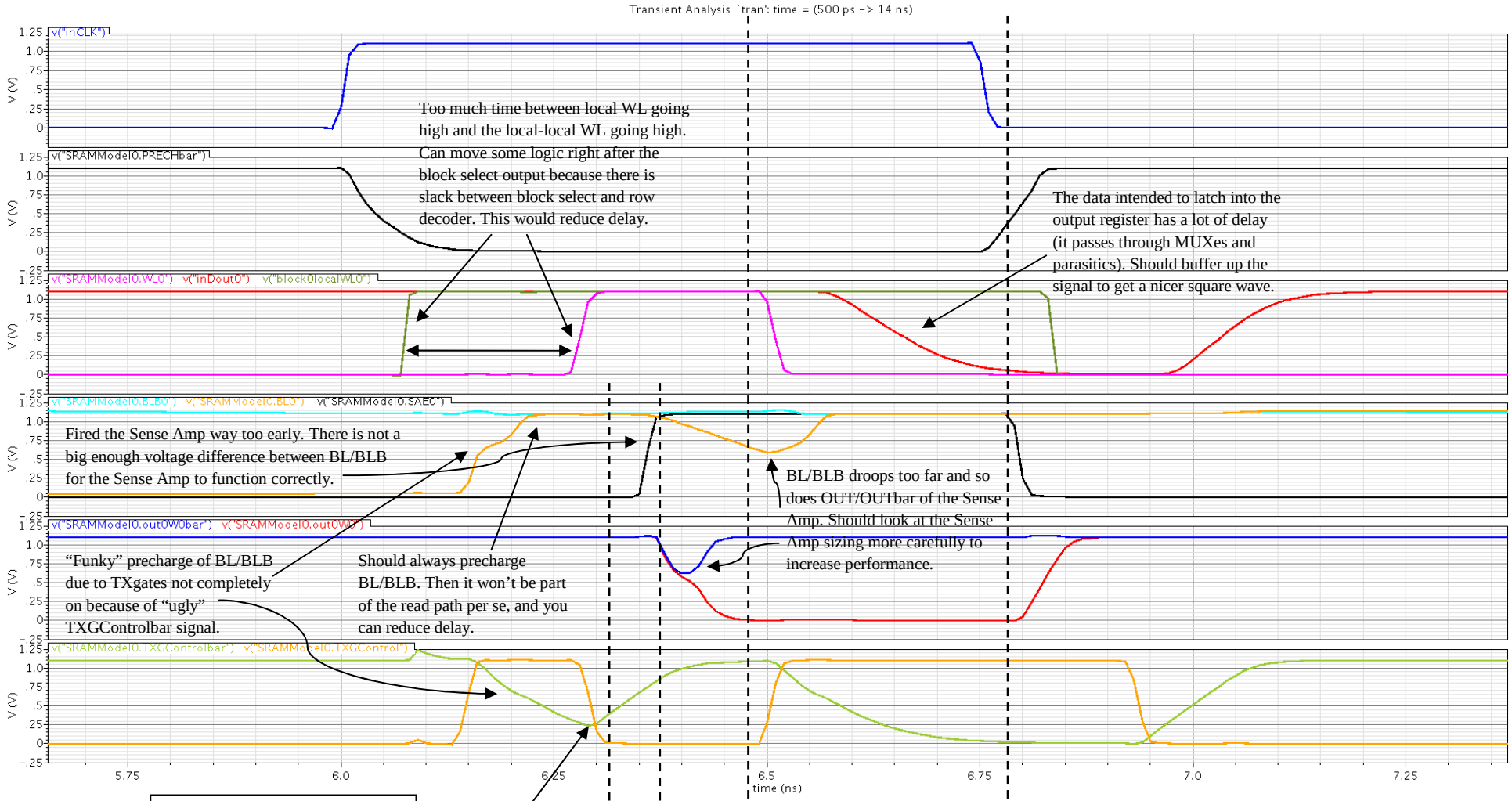

Too much time between local WL going high and the local-local WL going high. Can move some logic right after the block select output because there is slack between block select and row decoder. This would reduce delay.

The data intended to latch into the output register has a lot of delay (it passes through MUXes and parasitics). Should buffer up the signal to get a nicer square wave.

Fired the Sense Amp way too early. There is not a big enough voltage difference between BL/BLB for the Sense Amp to function correctly.

BL/BLB droops too far and so does OUT/OUTbar of the Sense Amp. Should look at the Sense Amp sizing more carefully to increase performance.

"Funky" precharge of BL/BLB due to TXgates not completely on because of "ugly" TXGControlbar signal.

Should always precharge BL/BLB. Then it won't be part of the read path per se, and you can reduce delay.

The reason BL or BLB doesn't begin to droop as soon as TXGControl (orange plot) goes low. The PMOS in the TXgate is still on (Fight between BL/BLB and BL/BLB driver – this is why BL or BLB doesn't begin to droop earlier). TXGControlbar (green) should be a perfect complement of TXGControl (orange) for proper functionality.

Because the output of the Sense Amp is valid at the left dotted line, if we buffer up the output path (i.e. make a nicer/quicker inDout0 signal) we can shorten the clock cycle, thus reducing our delay.

Too much time between TXgates turning off and BL or BLB drooping. BL or BLB should begin to droop as soon as TXGControl (orange in graph 6) goes low because BL/BLB float when TXGControl goes low.

For discussion purposes,
Top plot is graph 1 and bottom plot is graph 5
For clarity:
Graph 1 shows the CLK signal being used by the entire memory
Graph 2 shows PRECHbar signal used to precharge BL/BLB
Graph 3 shows:

- a local WL in green, which is created by the row decoder output ANDed with the block select output
- a local-local WL in pink, which is the local WL(green) passed through some logic to create control signals
- inDout0 in red, which is data bit 0 to be latched into the output register on the next falling CLK edge

Graph 4 shows BL in aqua and BLB in orange; Sense Amp Enable is in black
Graph 5 shows the output of the sense amp; OUT is red and OUTbar is blue
Graph 6 shows TXGControl in orange and TXGControlbar in green. These two signals control the transmission gate between the BL/BLB drivers and the actual BL/BLB.

Our measured delay was 0.7425 ns, which resulted in a CLK period of 1.425 ns.

- By buffering up the inDout signal (correct data from the specific block being sent through 2:1MUXes to the output register), we could've reduced our delay by ~150-200 ps alone.
- By just buffering up PRECHbar, we could've reduced our delay by ~100 ps.
- If we precharged after any given operation (when the CLK is low) we could've removed the precharge delay from our critical path completely and reduced our delay by ~200 ps. Then we would only worry about the local WL (green in graph 3) to dictate the beginning path of our read operation (as opposed to including precharging in the read operation).
- By examining the Sense Amp sizing more carefully, we could reduce the (voltage) amount BL/BLB and OUT/OUTbar drops, consequently reducing our delay by ~50 ps (or possibly more).

Combining these techniques together, we could have cut our delay in half, reducing our metric by 75%.